

WHAT IS CLAIMED IS:

1. A method for scheduling execution of one or more processes in a computer, comprising the steps of:
 - providing a timer;
 - providing a series of events for each process of a preselected set of processes, the events comprising a start time for each process;
 - starting execution of each process based on a time out of the timer, each process starting execution according to the corresponding start time;
 - providing a table;
 - storing each event in the table; and
 - operating the timer to be set for a time out at each of a series of reload values, each reload value being equal to a number of time increments until a next event in the table.
2. The method according to claim 1 wherein the events further include a deadline for each process of the preselected set of processes;
 - based on a time out of the timer, stopping execution of an executing process regardless of whether the process has stopped execution normally, each process stopping execution according to the corresponding deadline.
3. The method of claim 1, wherein if one process starts execution while another process is executing, preempting the process already executing.
4. The method of claim 1 wherein each process comprises one of a task and an ISR.
5. The method of claim 4 wherein when a process comprises an ISR, upon execution of the ISR, providing an enable time.
6. The method of claim 5 comprising the further step of disabling the ISR after

execution; and enabling the ISR upon expiration of the enable time.

7. A computer system comprising:

a timer;

a table storing a series of events for each process of a preselected set of processes, the events comprising a start time for each process;

an operating system causing execution of each process based on a time out of the timer, each process starting execution according to the corresponding start time stored in the table, the timer being arranged and configured to be set for a time out at each of a series of reload values, each reload value being equal to a number of time increments until a next event in the table.

8. The computer system of claim 7 wherein the table further stores events comprising a deadline for each process, the operating system causing, based on a time out of the timer, execution of an executing process to stop regardless of whether the process has stopped execution normally, each process stopping execution according to the corresponding deadline stored in the table.

9. The computer system of claim 8, wherein each process comprises one of a task and an ISR.

10. The computer system of claim 9 wherein when a process comprises an ISR, upon execution of the ISR, using an enable time in the table, the enable time causing disablement of an ISR after execution; and enablement of the ISR upon expiration of the enable time.

11. A method for scheduling one or more processes comprising the steps of:

providing a timer;

starting a plurality of processes based on a time out of the timer, each process starting execution according to a start time specified in a time table;

if one of the processes starts execution while another process is executing, preempting the process already executing;

if one of the processes has been preempted and the process that preempted the process stops execution, resuming the process that has been preempted; and

based on a time out of the timer, stopping execution of the processes regardless of whether the process has stopped execution normally, each process stopping execution according to a deadline specified in the time table;

setting the timer for a time out at each of a series of reload values, each reload value being equal to a number of time increments until a next one of a start time and deadline in the time table.

12. A method for scheduling one or more processes comprising the steps of:

providing a plurality of timers;

starting a plurality of processes based on a time out of a first one of the timers, each process starting execution according to a start time specified in a time table; and

based on a time out of a second one of the timers, stopping execution of the processes regardless of whether the process has stopped execution normally, each process stopping execution according to a deadline specified in the time table;

setting the first one of the timers for a time out at each of a series of reload values, each reload value being equal to a number of time increments until a next start time in the time table;

setting the second one of the timers for a time out at each of a series of reload values, each reload value being equal to a number of time increments until a next deadline in the time table.

13. The method according to claim 12 wherein each process comprises one of a task and an ISR, and comprising the further steps of when a process comprises an ISR, upon execution of the ISR, providing an enable time, disabling the ISR after execution; and enabling the ISR upon expiration of the enable time, the step of enabling being performed by setting a third one of the timers to time out after a reload value equal to

the enable time.

14. A computer system comprising:

a timer mechanism;

a table storing a series of events for each process of a preselected set of processes, the events comprising a start time for each process and a deadline for each process;

an operating system causing execution of each process based on a time out of the timer mechanism, each process starting execution according to the corresponding start time stored in the table and stopping execution according to the corresponding deadline stored in the table, the timer mechanism being arranged and configured to be set for a time out at each of a series of reload values, each reload value being equal to a number of time increments until a next event in the table;

the timer mechanism comprising a first timer set for a time out at each of a series of reload values, each reload value being equal to a number of time increments until a next start time in the time table, and a second timer set for a time out at each of a series of reload values, each reload value being equal to a number of time increments until a next deadline in the time table.

15. The computer system of claim 14 wherein each process comprises one of a task and an ISR, and further comprising a third timer to time out after a reload value equal to an enable time wherein the enable time is set, when a process comprises an ISR, upon execution of the ISR, the ISR being disabled after execution; and enabled upon time out of the third timer at the expiration of the enable time.